Machine Learning

Machine learning is a general term which specifies that if a machine can learn from data.

In Machine Learning you provide input, a result and a then a algorithm is applied to it to give a output.

Steps to Machine learning.

1. Data is collected
2. Data is pre-processed

* Analysing the data (seeing what would be useful for the model)
* Dealing the missing, nan data, handling duplicate data, handling abnormal data handling, outliers, extracting relevant feature, scaling data, and other pre-processing techniques in real life.
* Splitting the dataset into training and testing set

1. Model Creation.

* Choosing a learning algorithm to build a model.
* Training the model with the training set.
* Testing the trained model with the testing set

1. Finally, with the results from the testing the model the results are analysed and evaluated.

* Positive improvement required is implemented
* such as:
  + - * + collecting more data
        + adding or removing data or other features
        + choosing another algorithm.
        + Turning hyperparameter etc to improve the model performance.

A good machine learning model should predict the future.

Machine learning algorithm types.

* Supervised
* Each data point is labelled or associated with a category or values of interest.
* The goal of supervised learning is to study many labels like this and then to be able to make prediction about future data points

There are two types of supervised learning.

1. Classification

* Output variable is categorised to two or more classes.

1. Regression

* Output variable is real or continues value.
* Unsupervised
* Data points have no labels associated with them.
* The data needs to be organised in some way that its structure can be described.
* Unsupervised learning groups the data into clusters
* It finds different ways of looking at complex data to make it appear simpler.
* The mechanism of unsupervised learning is to fit the model with an example input without the associated output and repeat the step many times until the model drives structure from the data.
* Now with brand new input the model will find hidden structures in the data.

There are two subcategories of unsupervised learning.

1. Clustering

* Split group data to clusters that are similar or related to each other.

1. Association

* Find structure in a chaotic environment.
* Reinforcement learning
* In reinforcement leaning a algorithm gets to choose an action in response to each data point.
* The algorithm received feed back on how good the action was
* Based on this the algorithm changes its action to receive the highest reward.

Reading a confusion matrix

Attributes that have a high correlation with the label should be kept as they are adding value to our model.

A picture containing treemap chart

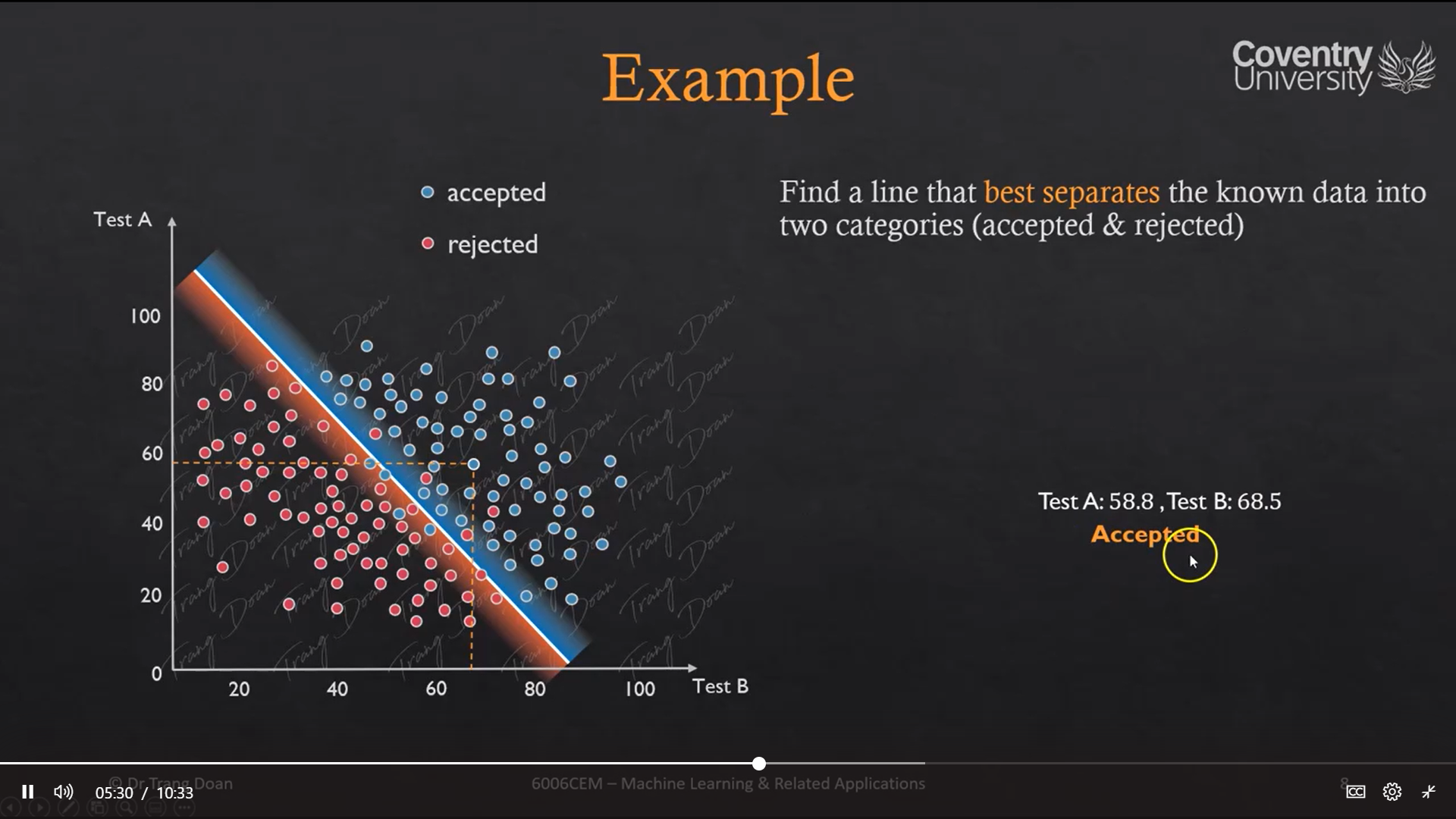
Description automatically generated

Attributes that have a low correlation to our model can be removed as they are not adding value to our model.

Attributes that have a high correlation with each other and not the label can be removed.

As they do not add value to our model.

**Logistic regression**



Drawing a line between the accepted and rejected data we can conclude which data will be most likely accepted and which data will most likely be rejected.

In the above example all the blue dots will be accepted and the red dots will be rejected.

**Neural networks**

10

Input

77

Out put = 0

else

Output = 1

Then

Threshold

If sum of the weight of input greater then

0

Threshold

10

Out put

**Graphical user interface

Description automatically generatedMultilayer neural network**

Acutual vs reidcted result example

Has output signals or classificationsto which input patterns may mark.

Neurons are in interconnected layers

**Graphical user interface

Description automatically generated**

So total error is calculated and fit backward to the networks that caused the weights and biases of the current model to change to minimize the error for each output.

It is repeated for every single sample of the training set to. Ultimately that a train that model that hopefully generalized enough to predict new unseen image of handwritten digits.

In each forward pass a backwards pass takes place

All sigmoid functions have the property that they map the entire number line into a small range such as between 0 and 1, or -1 and 1,

Each weighted sum has a bias before is added to the sigmoid function

Back work propagation

Forward propagation

Matches results with actual results

If correct positively is reinforced.

Otherwise it checks if the vaue is obtained is lower or higher then value it obtained and keeps looping to the input layer

Checks bias from results from previous neurons calculations

Hidden layers fine tune the input weighting until the neural networks margin of error is minimal

Collect input patterns

**Evaluating a model**

overfitting occurs when you fit the model to close to theparticularities of the training set.

While a too simple model is not good as it can’t represent the trend of the data and hence performs poorly on testing and training set. This ais a underfitted of high bias model.

**Diagram

Description automatically generated**Timeline

Description automatically generated with medium confidenceThe model in between these two is a sweet spot model as it performs well on testing and training data.

How to avoide under fitting and over fitting

1. Getting more data to train the model. If the training set is big enough o cover the trend of the data, it hep to better generilize and have more accurate prediction on new unseen data.
2. Try a **smaller set of features** to prevent overfitting. It can be done by our own choice or by applying mosle selection techniques or maybe **getting more features** might be required.
3. Getting more polonomial features.
4. Increasing or decreasing the learing rate.
5. Regulization, keeping all features but reducing the magnitude or the values of the parameter’s status. **This methoud works well when they are a lot of features**. If each of them are useful and each of them contributes to predicting the value of y. so they cant be thrown away.

If the testing and training error is low, then our model is good.

If the training low is high but the testing error is high, then the model is over fitted.

If the training error is high and the testing error is high then the model is underfitted,

**Cross validation**

in k fold cross validation, it is firstly decided how many equal blocks we want to divide the data into.

K – 1 blocks are used for training and k 1 bocks are used for testing. k[:-1], k[1]

When 75% of the data is used for training and 25 % of the data is used for testing its is called forfold cross validation.

**A picture containing chart

Description automatically generated**

Blocks are tranined and tested in variation and the scores are recorded.

The result of the k fold kross validation is the average of each set.

As we are using average from performing cross validation on one dataset using different models. The model with the highest cross validation result is the best model.

The best model has the highest testing score.

So we are generally cross validating models with each other

**Confusion matrix part 2**

Graphical user interface

Description automatically generated

**Chart

Description automatically generatedGraphical user interface, text, application

Description automatically generated**

**Exam notes to get better marks from Dr Trang**

Restart and clear output from Jupiter note book kernel so your course work could be valid

Keep the header names because you will need them latter to get access to the value of each column.

Not keeping them is not a good idea.

Keep the sample size above 1000 samples.

We are trying to generalize what the future of something should be

It is bad practice to use label encoder on features

Its best to use one hot encoder try to use one hot encoder and don’t use dummies or label encoder

Main reason in reality after you deploy a built model and if you have new data and the data has new values for example if the values are 4 and 5 so our model wasn’t trained using that model so if your are using get dummies then our model will fall.

Model accuracy is not always the best way to predict a models performance